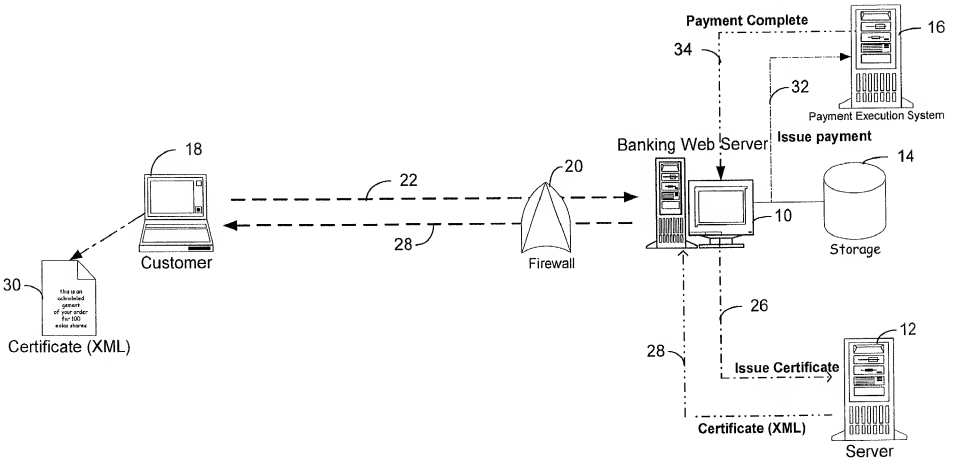

Payment Complete
16
34
32
Payment Execution System
Issue payment
Banking Web Server
14
18
22
20
10
Storage
Customer
28
Firewall
30
This is an
scheduled
payment
of your order
for 100
extra shares
Certificate (XML)
26
12
28
Issue Certificate
Certificate (XML)
Server

FIG. 1

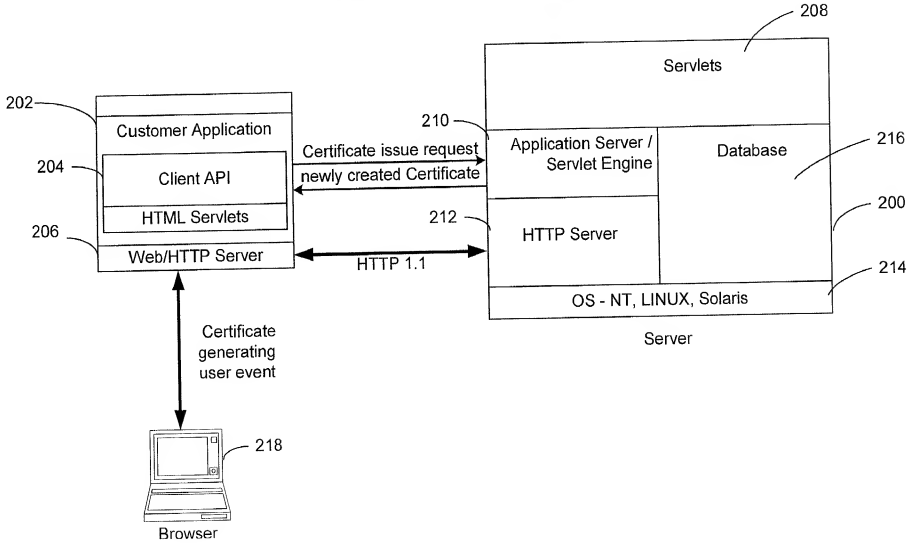
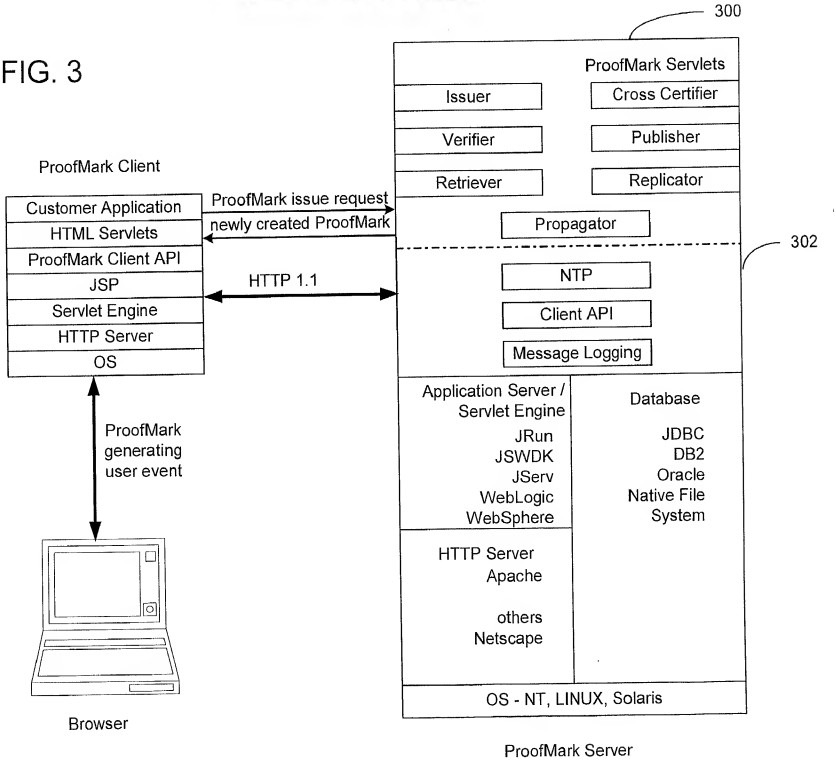

208
Servlets
202
Customer Application
210
Application Server / Servlet Engine
Database
216
204
Client API
Certificate issue request
newly created Certificate
HTML Servlets
212
HTTP Server
200
206
Web/HTTP Server
HTTP 1.1
214
OS - NT, LINUX, Solaris
Server
Certificate generating user event
218
Browser

FIG. 2

FIG. 3


300
ProofMark Servlets
Issuer
Cross Certifier
Verifier
Publisher
Retriever
Replicator
Propagator
302
NTP
Client API
Message Logging
Application Server / Servlet Engine
JRun
JSWDK
JServ
WebLogic
WebSphere
Database
JDBC
DB2
Oracle
Native File System
HTTP Server
Apache
others
Netscape
OS - NT, LINUX, Solaris
ProofMark Server
ProofMark Client
Customer Application
HTML Servlets
ProofMark Client API
JSP
Servlet Engine
HTTP Server
OS
ProofMark issue request
newly created ProofMark
HTTP 1.1
ProofMark generating user event
Browser

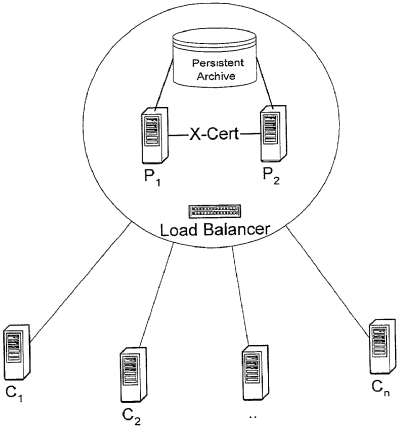


FIG. 4

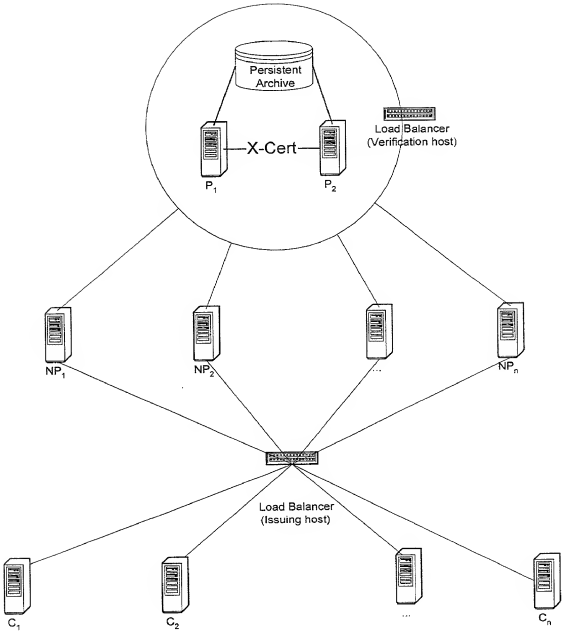

Persistent Archive
X-Cert
P1
P2
Load Balancer (Verification host)
NP1
NP2
...
NPn
Load Balancer (Issuing host)
C1
C2
...
Cn

FIG. 5

Printed Document

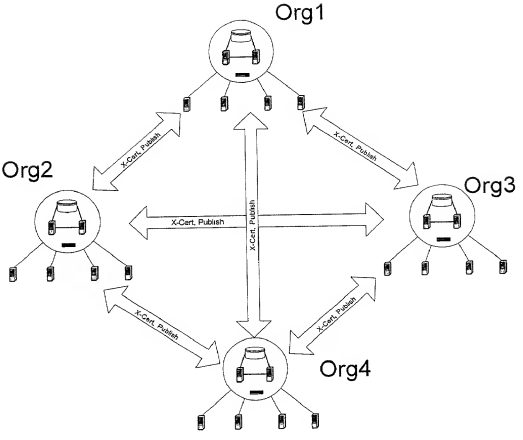


FIG. 6

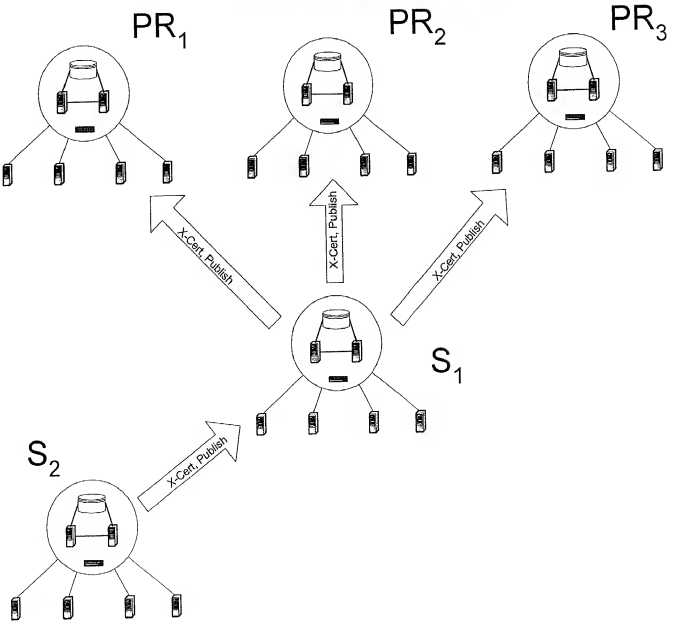

PR1
PR2
PR3
X-Cert. Publish
X-Cert. Publish
X-Cert. Publish
S1
S2
X-Cert. Publish

FIG. 7

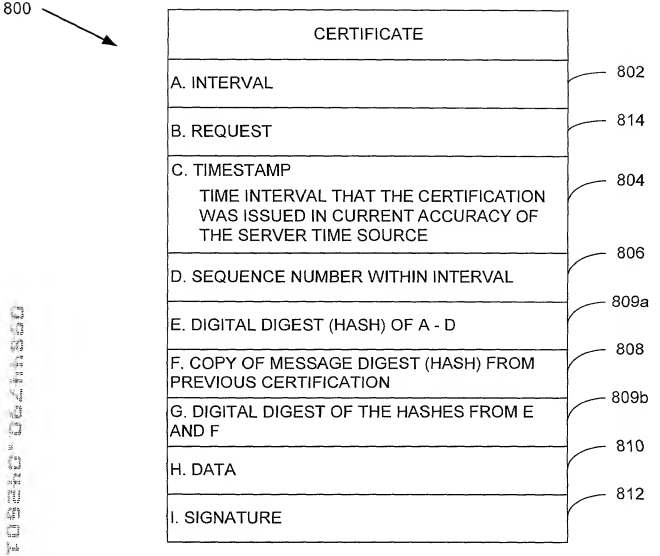

800
CERTIFICATE
A. INTERVAL
802
B. REQUEST
814
C. TIMESTAMP
TIME INTERVAL THAT THE CERTIFICATION WAS ISSUED IN CURRENT ACCURACY OF THE SERVER TIME SOURCE
804
D. SEQUENCE NUMBER WITHIN INTERVAL
806
E. DIGITAL DIGEST (HASH) OF A - D
809a
F. COPY OF MESSAGE DIGEST (HASH) FROM PREVIOUS CERTIFICATION
808
G. DIGITAL DIGEST OF THE HASHES FROM E AND F
809b
H. DATA
810
I. SIGNATURE
812

FIG. 8

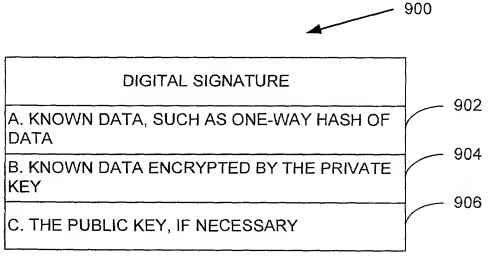

900
DIGITAL SIGNATURE
A. KNOWN DATA, SUCH AS ONE-WAY HASH OF DATA
902
B. KNOWN DATA ENCRYPTED BY THE PRIVATE KEY
904
C. THE PUBLIC KEY, IF NECESSARY
906

FIG. 9

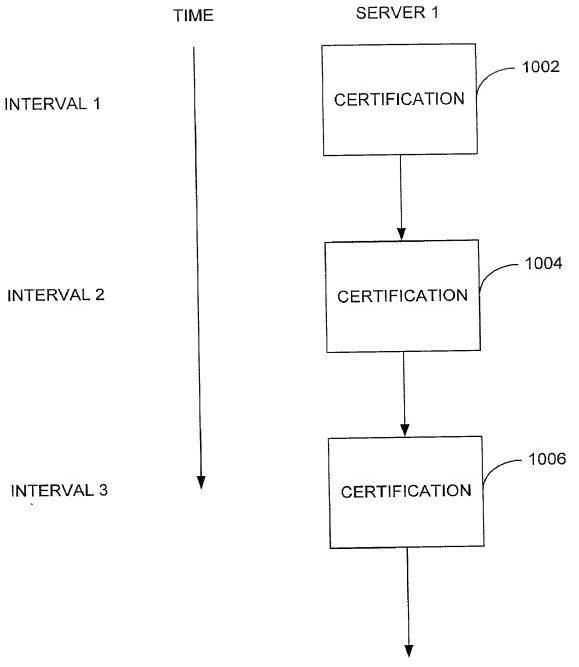

TIME
SERVER 1
INTERVAL 1
INTERVAL 2
INTERVAL 3
CERTIFICATION
CERTIFICATION
CERTIFICATION
1002
1004
1006

FIG. 10

Applicant's Exhibit 2

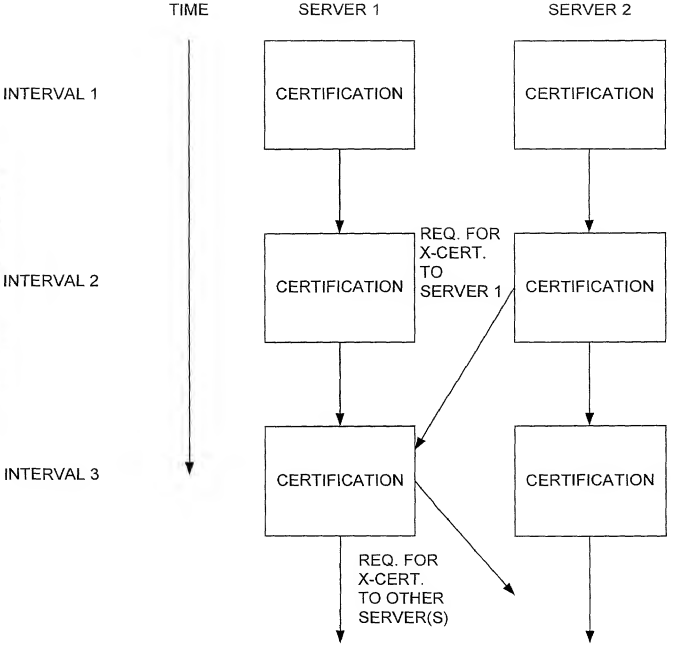


FIG. 11

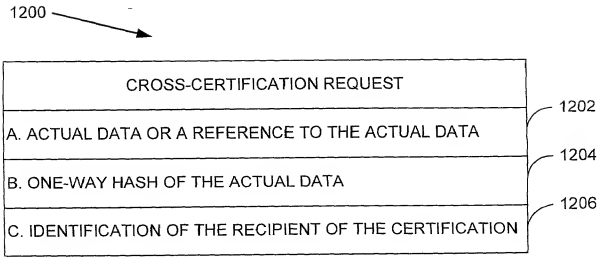

1200
CROSS-CERTIFICATION REQUEST
A. ACTUAL DATA OR A REFERENCE TO THE ACTUAL DATA
B. ONE-WAY HASH OF THE ACTUAL DATA
C. IDENTIFICATION OF THE RECIPIENT OF THE CERTIFICATION
1202
1204
1206

FIG. 12

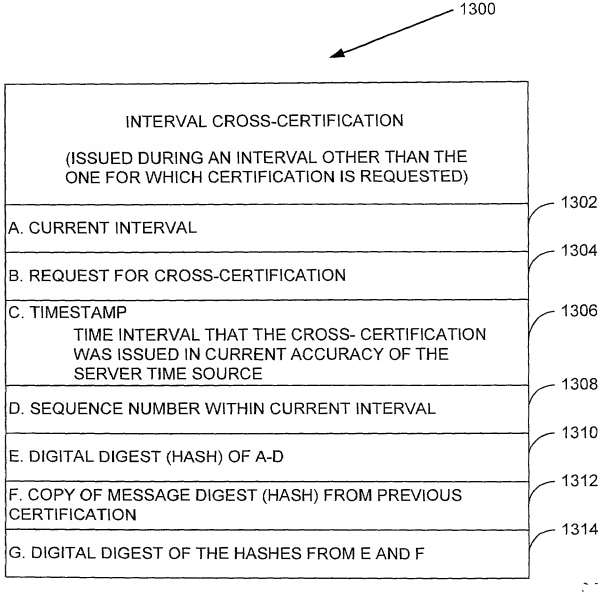

1300
INTERVAL CROSS-CERTIFICATION
(ISSUED DURING AN INTERVAL OTHER THAN THE ONE FOR WHICH CERTIFICATION IS REQUESTED)
A. CURRENT INTERVAL
B. REQUEST FOR CROSS-CERTIFICATION
C. TIMESTAMP
TIME INTERVAL THAT THE CROSS- CERTIFICATION WAS ISSUED IN CURRENT ACCURACY OF THE SERVER TIME SOURCE
D. SEQUENCE NUMBER WITHIN CURRENT INTERVAL
E. DIGITAL DIGEST (HASH) OF A-D
F. COPY OF MESSAGE DIGEST (HASH) FROM PREVIOUS CERTIFICATION
G. DIGITAL DIGEST OF THE HASHES FROM E AND F
1302
1304
1306
1308
1310
1312
1314

FIG. 13

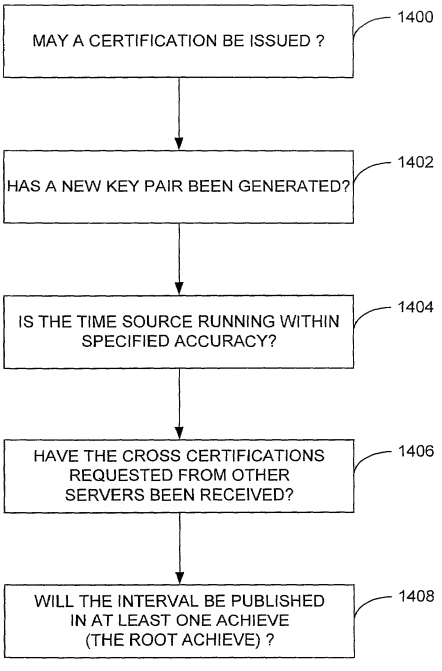

MAY A CERTIFICATION BE ISSUED ?
1400
HAS A NEW KEY PAIR BEEN GENERATED?
1402
IS THE TIME SOURCE RUNNING WITHIN SPECIFIED ACCURACY?
1404
HAVE THE CROSS CERTIFICATIONS REQUESTED FROM OTHER SERVERS BEEN RECEIVED?
1406
WILL THE INTERVAL BE PUBLISHED IN AT LEAST ONE ACHIEVE (THE ROOT ACHIEVE) ?
1408

FIG. 14

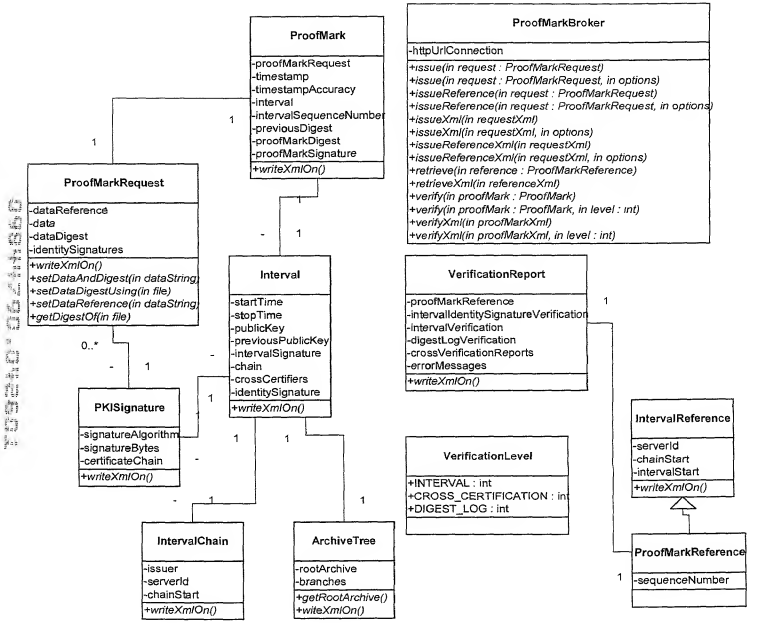

ProofMark
-proofMarkRequest
-timestamp
-timestampAccuracy
-interval
-intervalSequenceNumber
-previousDigest
-proofMarkDigest
-proofMarkSignature
+writeXmlOn()
ProofMarkBroker
-httpUrlConnection
+issue(in request : ProofMarkRequest)
+issue(in request : ProofMarkRequest, in options)
+issueReference(in request : ProofMarkRequest)
+issueReference(in request : ProofMarkRequest, in options)
+issueXml(in requestXml)
+issueXml(in requestXml, in options)
+issueReferenceXml(in requestXml)
+issueReferenceXml(in requestXml, in options)
+retrieve(in reference : ProofMarkReference)
+retrieveXml(in referenceXml)
+verify(in proofMark : ProofMark)
+verify(in proofMark : ProofMark, in level : int)
+verifyXml(in proofMarkXml)
+verifyXml(in proofMarkXml, in level : int)
ProofMarkRequest
-dataReference
-data
-dataDigest
-identitySignatures
+writeXmlOn()
+setDataAndDigest(in dataString)
+setDataDigestUsing(in file)
+setDataReference(in dataString)
+getDigestOf(in file)
Interval
-startTime
-stopTime
-publicKey
-previousPublicKey
-intervalSignature
-chain
-crossCertifiers
-identitySignature
+writeXmlOn()
VerificationReport
-proofMarkReference
-intervalIdentitySignatureVerification
-intervalVerification
-digestLogVerification
-crossVerificationReports
-errorMessages
+writeXmlOn()
PKISignature
-signatureAlgorithm
-signatureBytes
-certificateChain
+writeXmlOn()
VerificationLevel
+INTERVAL : int
+CROSS_CERTIFICATION : int
+DIGEST_LOG : int
IntervalReference
-serverId
-chainStart
-intervalStart
+writeXmlOn()
IntervalChain
-issuer
-serverId
-chainStart
+writeXmlOn()
ArchiveTree
-rootArchive
-branches
+getRootArchive()
+writeXmlOn()
ProofMarkReference
-sequenceNumber
1
0..*

FIG. 15

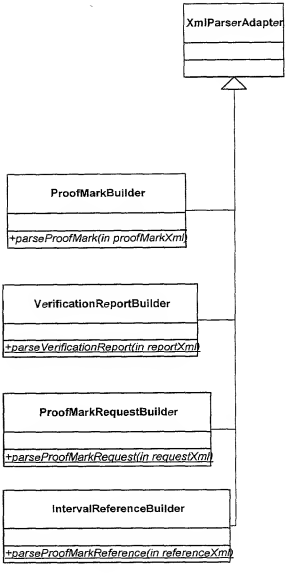

XmlParserAdapter
ProofMarkBuilder
+parseProofMark(in proofMarkXml)
VerificationReportBuilder
+parseVerificationReport(in reportXml)
ProofMarkRequestBuilder
+parseProofMarkRequest(in requestXml)
IntervalReferenceBuilder
+parseProofMarkReference(in referenceXml)

FIG. 16

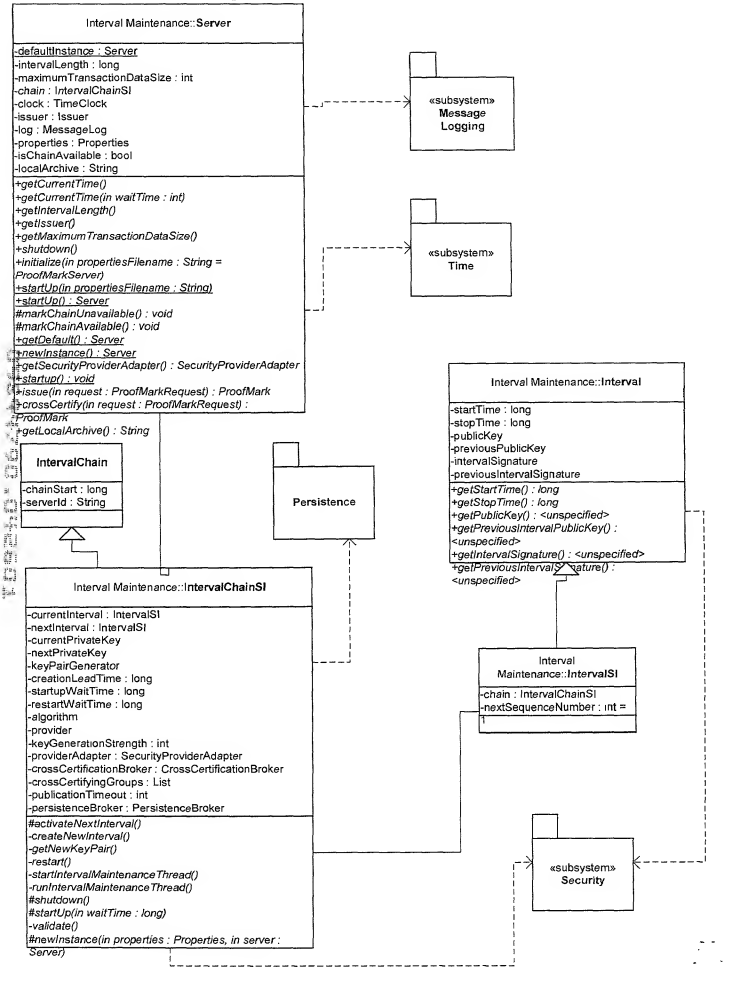

Interval Maintenance::Server
-defaultInstance : Server
-intervalLength : long
-maximumTransactionDataSize : int
-chain : IntervalChainSI
-clock : TimeClock
-issuer : Issuer
-log : MessageLog
-properties : Properties
-isChainAvailable : bool
-localArchive : String
+getCurrentTime()
+getCurrentTime(in waitTime : int)
+getIntervalLength()
+getIssuer()
+getMaximumTransactionDataSize()
+shutdown()
+initialize(in propertiesFilename : String = ProofMarkServer)
+startUp(in propertiesFilename : String)
+startUp() : Server
#markChainUnavailable() : void
#markChainAvailable() : void
+getDefault() : Server
+newInstance() : Server
+getSecurityProviderAdapter() : SecurityProviderAdapter
+startup() : void
+issue(in request : ProofMarkRequest) : ProofMark
+crossCertify(in request : ProofMarkRequest) : ProofMark
+getLocalArchive() : String
«subsystem» Message Logging
«subsystem» Time
Interval Maintenance::Interval
-startTime : long
-stopTime : long
-publicKey
-previousPublicKey
-intervalSignature
-previousIntervalSignature
+getStartTime() : long
+getStopTime() : long
+getPublicKey() : <unspecified>
+getPreviousIntervalPublicKey() : <unspecified>
+getIntervalSignature() : <unspecified>
+getPreviousIntervalSignature() : <unspecified>
IntervalChain
-chainStart : long
-serverId : String
Persistence
Interval Maintenance::IntervalChainSI
-currentInterval : IntervalSI
-nextInterval : IntervalSI
-currentPrivateKey
-nextPrivateKey
-keyPairGenerator
-creationLeadTime : long
-startupWaitTime : long
-restartWaitTime : long
-algorithm
-provider
-keyGenerationStrength : int
-providerAdapter : SecurityProviderAdapter
-crossCertificationBroker : CrossCertificationBroker
-crossCertifyingGroups : List
-publicationTimeout : int
-persistenceBroker : PersistenceBroker
#activateNextInterval()
-createNewInterval()
-getNewKeyPair()
-restart()
-startIntervalMaintenanceThread()
-runIntervalMaintenanceThread()
#shutdown()
#startUp(in waitTime : long)
-validate()
#newInstance(in properties : Properties, in server : Server)
Interval Maintenance::IntervalSI
-chain : IntervalChainSI
-nextSequenceNumber : int = 1
«subsystem» Security

FIG. 17

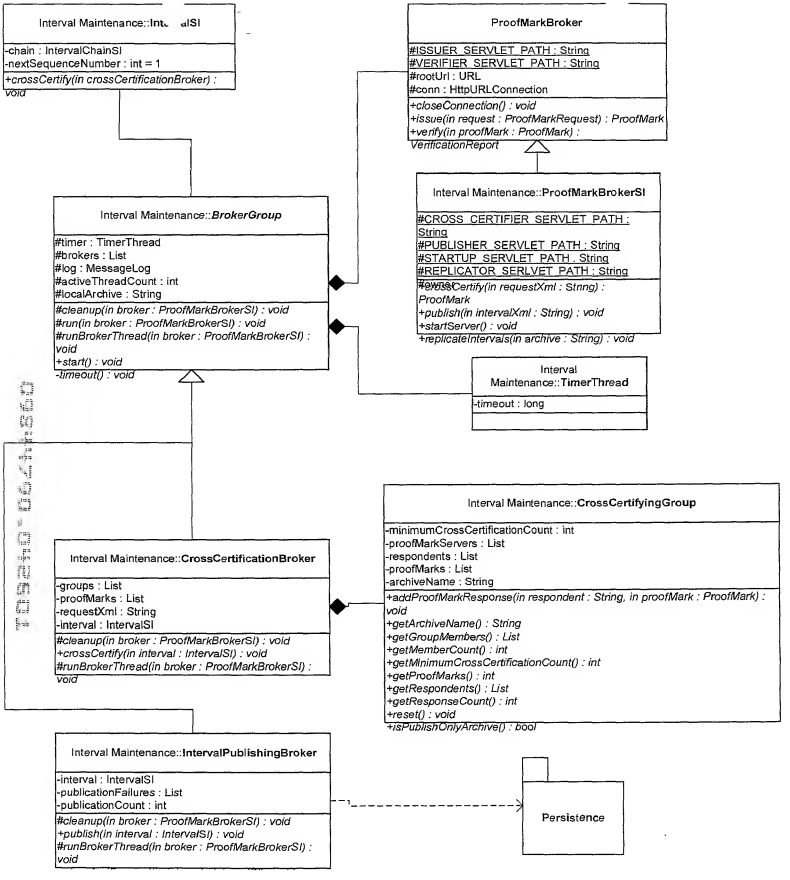

Interval Maintenance::IntervalSI
-chain : IntervalChainSI
-nextSequenceNumber : int = 1
+crossCertify(in crossCertificationBroker) : void
ProofMarkBroker
#ISSUER_SERVLET_PATH : String
#VERIFIER_SERVLET_PATH : String
#rootUrl : URL
#conn : HttpURLConnection
+closeConnection() : void
+issue(in request : ProofMarkRequest) : ProofMark
+verify(in proofMark : ProofMark) : VerificationReport
Interval Maintenance::BrokerGroup
#timer : TimerThread
#brokers : List
#log : MessageLog
#activeThreadCount : int
#localArchive : String
#cleanup(in broker : ProofMarkBrokerSI) : void
#run(in broker : ProofMarkBrokerSI) : void
#runBrokerThread(in broker : ProofMarkBrokerSI) : void
+start() : void
-timeout() : void
Interval Maintenance::ProofMarkBrokerSI
#CROSS_CERTIFIER_SERVLET_PATH : String
#PUBLISHER_SERVLET_PATH : String
#STARTUP_SERVLET_PATH : String
#REPLICATOR_SERLVET_PATH : String
+crossCertify(in requestXml : Strng) : ProofMark
+publish(in intervalXml : String) : void
+startServer() : void
+replicateIntervals(in archive : String) : void
Interval Maintenance::TimerThread
-timeout : long
Interval Maintenance::CrossCertificationBroker
-groups : List
-proofMarks : List
-requestXml : String
-interval : IntervalSI
#cleanup(in broker : ProofMarkBrokerSI) : void
+crossCertify(in interval : IntervalSI) : void
#runBrokerThread(in broker : ProofMarkBrokerSI) : void
Interval Maintenance::CrossCertifyingGroup
-minimumCrossCertificationCount : int
-proofMarkServers : List
-respondents : List
-proofMarks : List
-archiveName : String
+addProofMarkResponse(in respondent : String, in proofMark : ProofMark) : void
+getArchiveName() : String
+getGroupMembers() : List
+getMemberCount() : int
+getMinimumCrossCertificationCount() : int
+getProofMarks() : int
+getRespondents() : List
+getResponseCount() : int
+reset() : void
+isPublishOnlyArchive() : bool
Interval Maintenance::IntervalPublishingBroker
-interval : IntervalSI
-publicationFailures : List
-publicationCount : int
#cleanup(in broker : ProofMarkBrokerSI) : void
+publish(in interval : IntervalSI) : void
#runBrokerThread(in broker : ProofMarkBrokerSI) : void
Persistence

FIG. 18

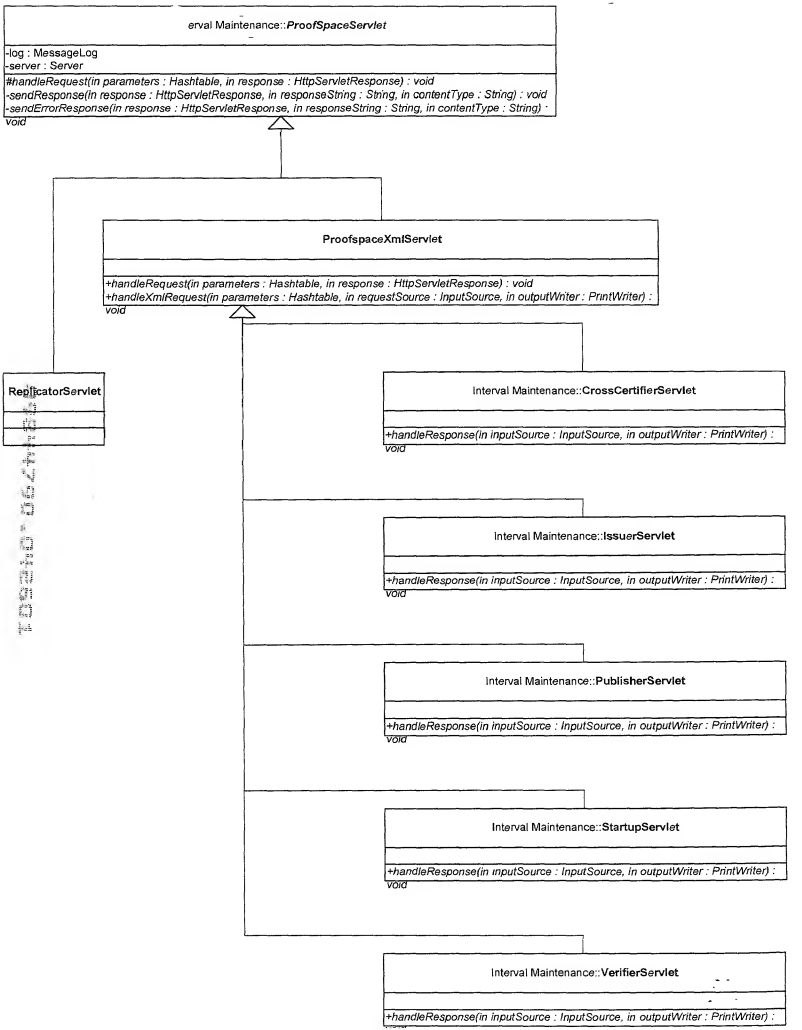

erval Maintenance::ProofSpaceServlet
-log : MessageLog
-server : Server
#handleRequest(in parameters : Hashtable, in response : HttpServletResponse) : void
-sendResponse(in response : HttpServletResponse, in responseString : String, in contentType : String) : void
-sendErrorResponse(in response : HttpServletResponse, in responseString : String, in contentType : String) :
void
ProofspaceXmlServlet
+handleRequest(in parameters : Hashtable, in response : HttpServletResponse) : void
+handleXmlRequest(in parameters : Hashtable, in requestSource : InputSource, in outputWriter : PrintWriter) :
void
ReplicatorServlet
Interval Maintenance::CrossCertifierServlet
+handleResponse(in inputSource : InputSource, in outputWriter : PrintWriter) :
void
Interval Maintenance::IssuerServlet
+handleResponse(in inputSource : InputSource, in outputWriter : PrintWriter) :
void
Interval Maintenance::PublisherServlet
+handleResponse(in inputSource : InputSource, in outputWriter : PrintWriter) :
void
Interval Maintenance::StartupServlet
+handleResponse(in inputSource : InputSource, in outputWriter : PrintWriter) :
void
Interval Maintenance::VerifierServlet
+handleResponse(in inputSource : InputSource, in outputWriter : PrintWriter) :

FIG. 19

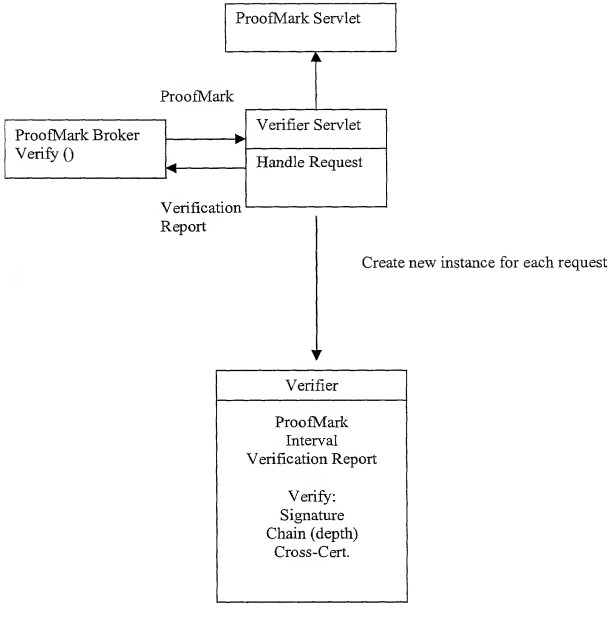

ProofMark Servlet
ProofMark
ProofMark Broker
Verify ()
Verifier Servlet
Handle Request
Verification
Report
Create new instance for each request
Verifier
ProofMark
Interval
Verification Report
Verify:
Signature
Chain (depth)
Cross-Cert.

FIG. 20

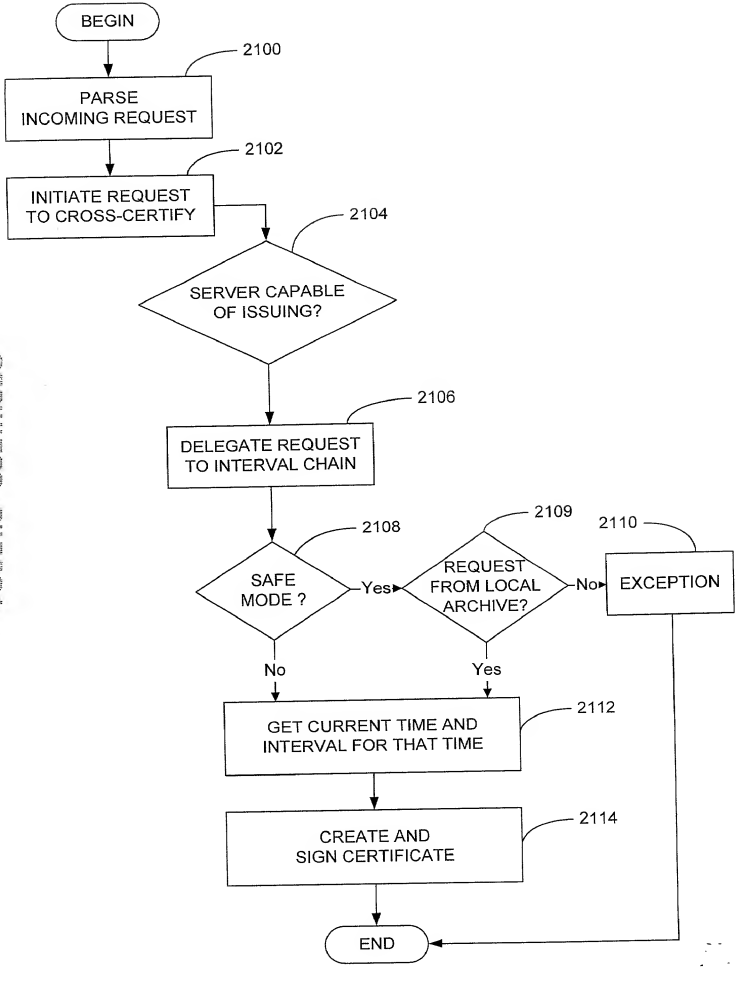

BEGIN
2100
PARSE INCOMING REQUEST
2102
INITIATE REQUEST TO CROSS-CERTIFY
2104
SERVER CAPABLE OF ISSUING?
2106
DELEGATE REQUEST TO INTERVAL CHAIN
2108
SAFE MODE ?
Yes
2109
REQUEST FROM LOCAL ARCHIVE?
No
2110
EXCEPTION
No
Yes
2112
GET CURRENT TIME AND INTERVAL FOR THAT TIME
2114
CREATE AND SIGN CERTIFICATE
END

FIG. 21

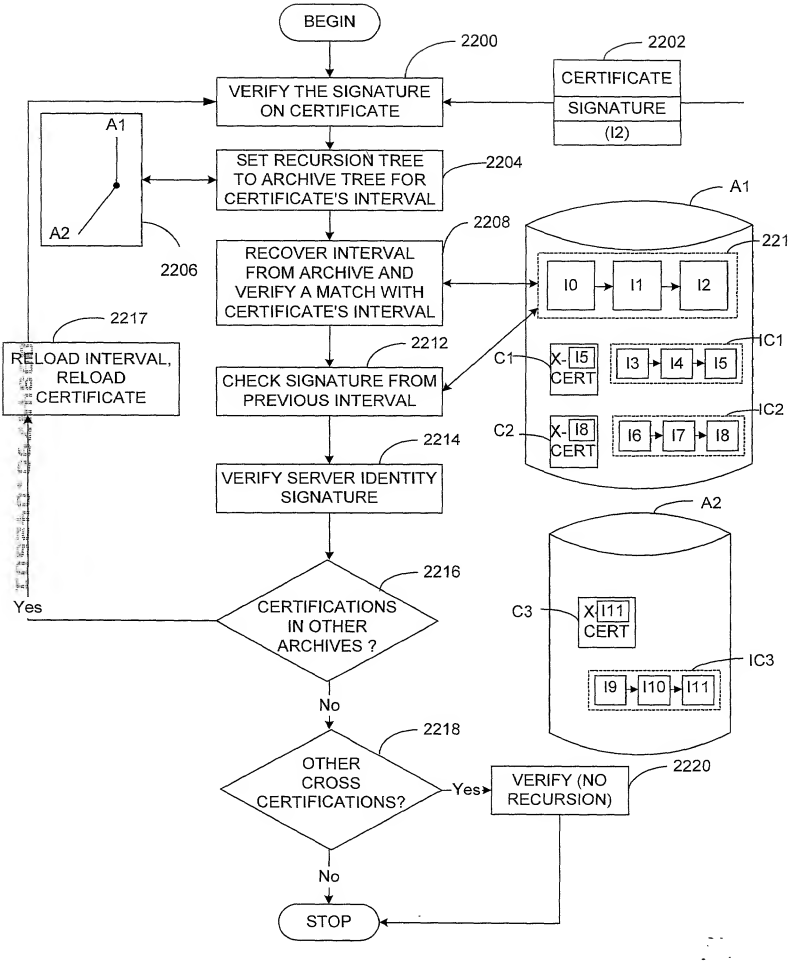

BEGIN
2200
VERIFY THE SIGNATURE ON CERTIFICATE
2202
CERTIFICATE
SIGNATURE
(I2)
A1
A2
2206
SET RECURSION TREE TO ARCHIVE TREE FOR CERTIFICATE'S INTERVAL
2204
RECOVER INTERVAL FROM ARCHIVE AND VERIFY A MATCH WITH CERTIFICATE'S INTERVAL
2208
A1
221
I0
I1
I2
2217
RELOAD INTERVAL, RELOAD CERTIFICATE
2212
CHECK SIGNATURE FROM PREVIOUS INTERVAL
C1
X-I5 CERT
I3
I4
I5
IC1
C2
X-I8 CERT
I6
I7
I8
IC2
2214
VERIFY SERVER IDENTITY SIGNATURE
A2
2216
CERTIFICATIONS IN OTHER ARCHIVES ?
Yes
C3
X-I11 CERT
I9
I10
I11
IC3
No
2218
OTHER CROSS CERTIFICATIONS?
Yes
VERIFY (NO RECURSION)
2220
No
STOP

FIG. 22

xxxx yyyy zzzz
xxxx yyyy zzzz
aaa bbb
aaa bbb
ccc ddd
ccc ddd
eee ffffff gggg
eee ffffff gggg
hhh
hhh
iiiiiiiii
iiiiiiiii
kkk
kkk

FIG. 23

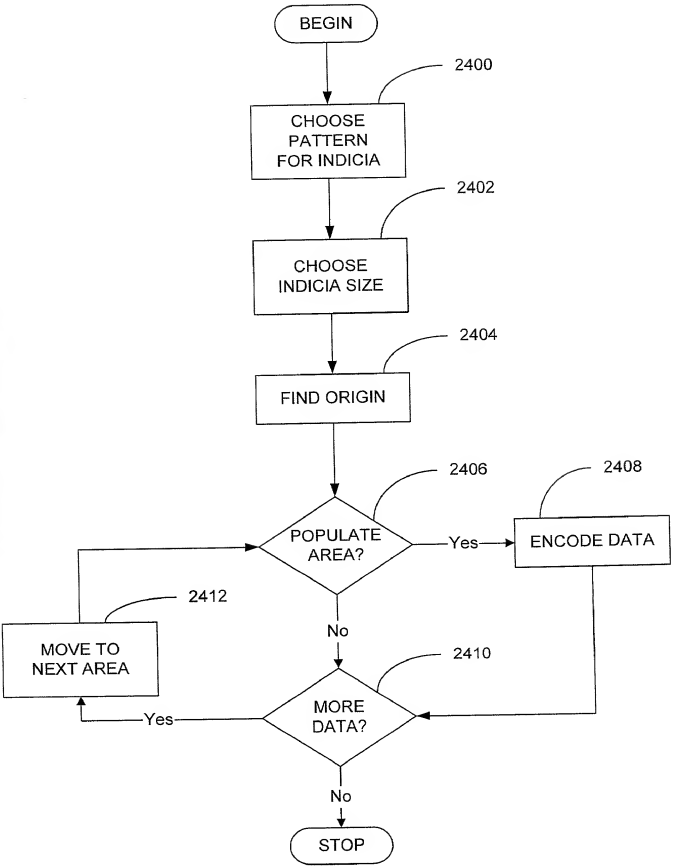

BEGIN
2400
CHOOSE PATTERN FOR INDICIA
2402
CHOOSE INDICIA SIZE
2404
FIND ORIGIN
2406
POPULATE AREA?
Yes
2408
ENCODE DATA
No
2412
MOVE TO NEXT AREA
2410
MORE DATA?
Yes
No
STOP

FIG. 24

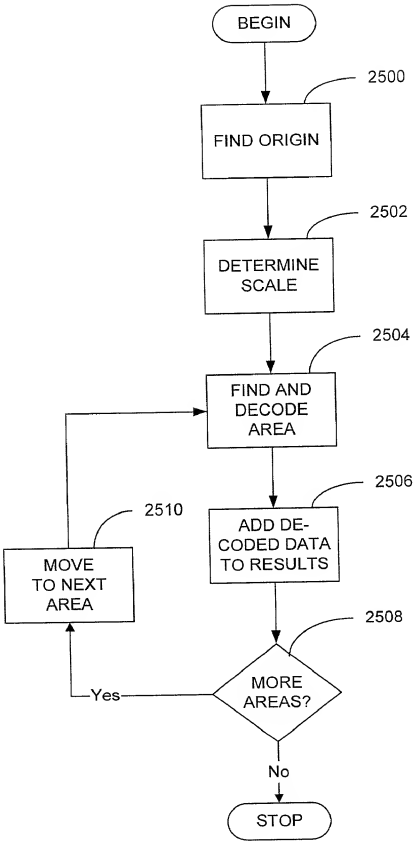

BEGIN
2500
FIND ORIGIN
2502
DETERMINE SCALE
2504
FIND AND DECODE AREA
2506
ADD DE-CODED DATA TO RESULTS
2508
MORE AREAS?
Yes
No
2510
MOVE TO NEXT AREA
STOP

FIG. 25

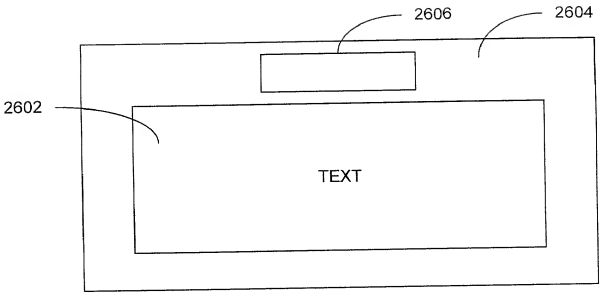


FIG. 26

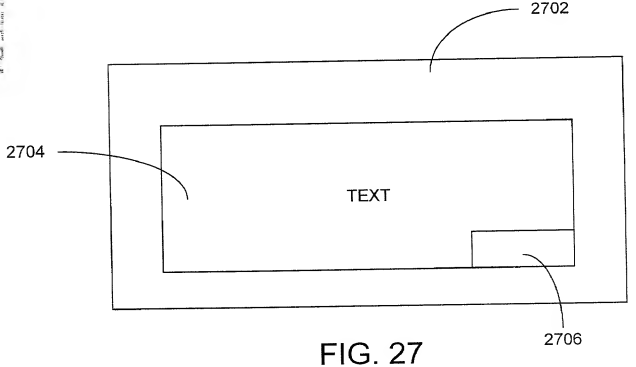


FIG. 27

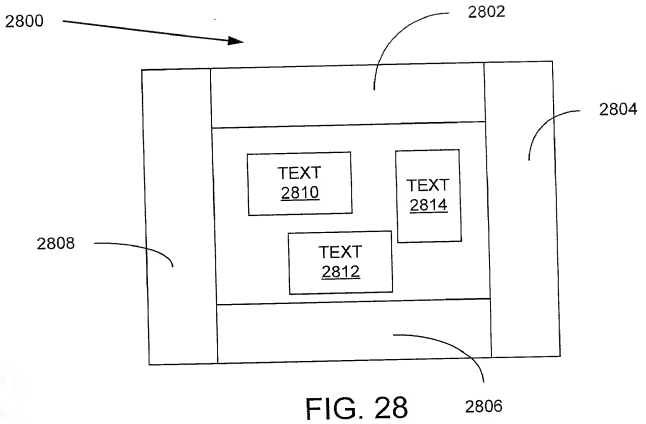


FIG. 28

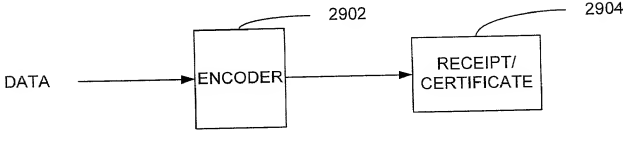


FIG. 29

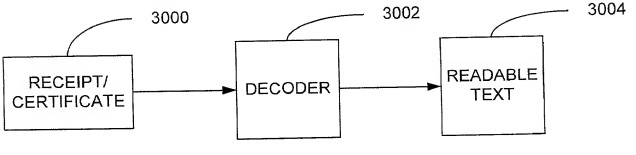


FIG. 30